

```

import cv2
import numpy as np
#Read an image
img = cv2.imread('flower.jpg')

#Display the image
cv2.imshow('Original Image', img)

#Get image dimensions (rows, columns, color channels)
[rows,cols,channels] = img.shape
print("Image size: " + str(rows) + " x " + str(cols) + " x " + str(channels))

#Check color model (grayscale or RGB)
if channels == 1:
    print('Color Model: Grayscale')
else:
    print('Color Model: RGB')

#Access individual pixels(example: center pixel)
center_row = rows//2 + 1
center_col = cols//2 + 1
center_pixel = img[center_row:center_col,:]
print("Center pixel: ", end="")
index = 2
while index >= 0:
    print(center_pixel[index],end=" ")
    index-=1

#Basic arithmetic operations(add constant value to all pixels)
brightness = np.ones(img.shape, dtype="uint8")*50
brightened_img = cv2.add(img,brightness)
cv2.imshow('Image Brightened', brightened_img)

#Basic geometric operation (flipping image horizontally)
flipped_img = cv2.flip(img, 1)
cv2.imshow('Image Flipped Horizontally', flipped_img)

#Get image channels
(blue,green,red) = cv2.split(img)
black = np.zeros(img.shape[:2],dtype="uint8")
img_red = cv2.merge([black,black,red])
img_green = cv2.merge([black,green,black])
img_blue = cv2.merge([blue,black,black])
img_final = cv2.hconcat([img_red,img_green,img_blue])
cv2.imshow('Channels', img_final)

#Darken the image
brightness = np.ones(img.shape, dtype="uint8")*50
darkened_img = cv2.subtract(img,brightness)
cv2.imshow('Image Darkened', darkened_img)
img_final = cv2.hconcat([img,darkened_img])
cv2.imshow('Image Darkened', img_final)

#Darken and brighten the image using multiplication and division
brightness = np.ones(img.shape, dtype="uint8")*2
brighten_img = cv2.multiply(img,brightness)
darken_img = cv2.divide(img,brightness)
img_final = cv2.hconcat([img,brighten_img,darken_img])
cv2.imshow('Image Brightened and Darkened', img_final)

#Rotates the image
img_rotate = cv2.rotate(img, cv2.ROTATE_90_COUNTERCLOCKWISE)
cv2.imshow('Original Image', img)
cv2.imshow('Image Rotated', img_rotate)

```

```
cv2.waitKey(0)  
cv2.destroyAllWindows()
```